自动化设计与技术欧洲会议(2022) **DOI：**10.23919/DATE54114.2022.9774698
Design, Automation and Test in Europe Conference (DATE 2022) 第四卷(期):975-980,2022 年

EVOLVE：大数据、高性能和云计算世界的融合

阿基里斯·泽内托普洛斯*，迪莫斯泰尼斯·马苏罗斯*，康斯坦蒂娜·科里奥乔治*，索蒂里奥斯·西迪斯*†，迪米特里奥斯·苏德里斯*，安东尼·查扎皮斯‡，克里斯托·科扎尼蒂斯‡，安杰洛斯·比拉斯‡，克里斯蒂安·平托，阮惠南§，斯泰利奥斯·卢卢达基斯，乔治斯·加尔迪基斯， 乔治·万瓦卡斯‡‡，米歇尔·奥布伦¶¶，克里斯蒂·西蒙尼杜§§，瓦西利斯·斯皮达基斯§§，康斯坦丁诺斯·西洛吉安诺普洛斯||，伯恩哈德·佩施尔||，塔希尔·埃姆雷·卡拉吉**，亚历山大斯托克**，让·托马斯·阿夸维瓦¶，

*希腊雅典通信与计算机系统研究所 (ICCS)
†希腊雅典哈罗科皮奥大学 (HUA) 信息学和远程信息处理系 (DIT)
‡计算机科学研究所，FORTH，希腊伊拉克利翁，lIBM 欧洲研究院，爱尔兰都柏林
§ATOS/Bull，法国巴黎，††Sunlight.io，希腊伊拉克利翁，‡Space Hellas S.A.，希腊雅典
¶¶Thales Alenia Space，法国图卢兹，§§NEUROCOM Luxembourg，卢森堡，||AVL List GmbH，格拉茨，奥地利
**Virtual Vehicle Research GmbH，奥地利格拉茨，¶DataDirect Networks，巴黎，法国

EVOLVE: Towards Converging Big-Data, High-Performance and Cloud-Computing Worlds

Achilleas Tzenetopoulos, Dimosthenis Masouros, Konstantina Koliogeorgi , Sotirios Xydis†, Dimitrios Soudris,
Antony Chazapis‡, Christos Kozanitis‡, Angelos Bilas‡, Christian Pintol, Huy-Nam Nguyen§, Stelios Louloudakis††,
Georgios Gardikis‡‡, George Vamvakas‡‡, Michelle Aubrun¶¶, Christy Symeonidou§§, Vassilis Spitadakis§§,
Konstantinos Xylogiannopoulos , Bernhard Peischl, Tahir Emre Kalayci, Alexander Stocker,Jean-Thomas Acquaviva¶,
*Institute of Communication and Computer Systems (ICCS), Athens, Greece
†Department of Informatics and Telematics (DIT), Harokopio University of Athens (HUA), Greece
‡Institute of Computer Science, FORTH, Heraklion, Greece, lIBM Research Europe, Dublin, Ireland
§ATOS/Bull, Paris, France, ††Sunlight.io, Heraklion, Greece, ‡‡Space Hellas S.A., Athens, Greece
¶¶Thales Alenia Space, Toulouse, France, §§NEUROCOM Luxembourg, Luxembourg, ||AVL List GmbH, Graz, Austria
**Virtual Vehicle Research GmbH, Graz, Austria, ¶DataDirect Networks, Paris, France.

Abstract EVOLVE is a pan European Innovation Action that aims to fully-integrate High-Performance-Computing (HPC) hardware with state-of-the-art software technologies under a unique testbed, that enables the convergence of HPC, Cloud and Big-Data worlds and increases our ability to extract value from massive and demanding datasets. EVOLVE's advanced compute platform combines HPC-enabled capabilities, with transparent deployment in high abstraction level, and a versatile Big-Data processing stack for

会议日期：2022 年 3 月 14-23 日 **会议地点：**比利时安特卫普
添加到 IEEE 的日期 *Xplore*：2022 年 5 月 19 日 **发布者：**IEEE
电子 ISBN：978-3-9819263-6-1 **按需打印 (PoD) ISBN：**978-1-6654-9637-7
电子 ISSN： 1558-1101 **按需打印 (PoD) ISSN：** 1530-1591

DOI：10.23919/DATE54114.2022.9774698
第四卷(期):975-980,2022 年

end-to-end workflows. Hence, domain experts have the potential to improve substantially the efficiency of existing services or introduce new models in the respective domains, e.g., automotive services, bus transportation, maritime surveillance and others. In this paper, we describe EVOLVE's testbed, and evaluate the performance of the integrated pilots from different domains.

Index Terms——dynamic outlier detection; heterogeneous information network; tensor representation; tensor index tree; clustering

摘要 EVOLVE 是一个泛欧洲创新行动计划，旨在将高性能计算（HPC）硬件与最先进的软件技术完全集成在一个独特的测试平台上，实现 HPC、云计算和大数据领域的融合，并提高我们从大规模复杂数据集中提取价值的能力。EVOLVE 先进的计算平台结合了 HPC 功能，在高级抽象层上实现了透明的部署，以及一个多功能的用于端到端工作流程的大数据处理堆栈。因此，领域专家有可能显著提高现有服务的效率，或者在相应的领域引入新的模型，例如汽车服务、公共汽车运输、海上监视等。在本文中，我们描述了 EVOLVE 的测试平台，并评估了来自不同领域的集成试点的性能。

索引术语 HPC、云计算、大数据、计算 平台、加速器、干扰、资源编排

1 介绍

随着数据成为现代经济和社会的创新中心，组织面临着新的挑战和限制。尽管在过去几年中，在提高商品系统的数据处理效率以及利用大数据和云技术提供新服务方面取得了巨大进展，但预计的数据洪流将企业、消费者和整个社会带到了一个新的前沿：我们如何处理需要苛刻计算的海量数据？

今天，在数据集大小和数据处理方面创建新的数据密集型服务是一个繁重且昂贵的过程，需要深厚的专业知识，即，高性能硬件、复杂的软件堆栈和专用的每个应用程序测试台，以实现所需的性能水平。然而，大多数组织，特别是中小型企业，缺乏这些资源和所需技能。EVOLVE 是一个泛欧洲的创新行动，旨在通过整合 HPC，大数据和云世界的技术来建立一个大规模的测试平台。EVOLVE 构建并演示了来自苛刻应用领域的真实生活，大量数据集的拟议平台，例如，汽车服务、海事监视等。

更具体地说，EVOLVE 集成了一个先进的、支持 HPC 的计算平台，能够处理大量要求苛刻的数据集，通过 HPC 功能丰富大数据处理，包括加速、大内存、快速存储架构和快速互连。它还使应用程序能够以工作流的形式表达其逻辑和数据集，这些工作流可以在没有大量 IT 专业知识的情况下跨领域专家组进行自动化，共享，改进和维护。此外，EVOLVE 提供了一个丰富而通用的大数据处理软件栈，可以通过使用大数据世界中的流行组件作为工作流阶段来执行自动化工作流。EVOLVE 使用基于云的方法，支持跨应用程序共享独特的测试平台，促进部署，访问和使用。最后，它提供了一个功能强大的测试平台，集成了项目技术，供生态系统利益相关者和相关方使用。

在本文中，我们走过的最终版本的 EVOLVE 的融合测试床。我们首先讨论和评估其硬件堆栈的基础上广泛使用的基准套件。然后，我们提出了核心的软件技术和他们的改进版本，采用透明的部署，优化和管理。然后对各种集成导频进行评估，显示其能够提供更高的性能效率，平均性能提高 26.3 倍，最高可达 114 倍。

本文的其余部分组织如下。在第 2 节中，我们通过描述和评估硬件堆栈的各个方面来展示我们的平台，这些硬件堆栈支持 HPC，云和大数据的融合。第 3 节，我们描述了系统和软件技术，这些技术可以有效地、抽象地利用上述硬件资源。第四部分的最后，我们将介绍这些技术。在第 4 节中，我们展示了来自不同领域的几个导向器的集成，并评估了它们在 EVOLVE 上部署时的性能。

2 EVOLVE 的硬件堆栈

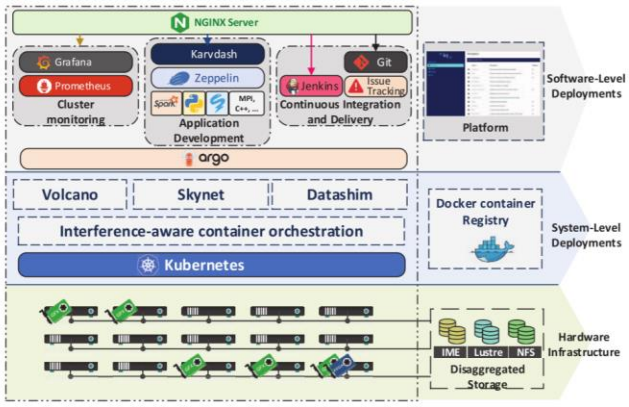


Fig. 1: Overview of EVOLVE platform

图 1：EVOLVE 平台概述

EVOLVE 提供高性能的硬件基础设施，并辅以紧密集成的系统和软件堆栈。图 1 总结了 EVOLVE 测试平台的集成技术，从硬件基础设施到系统级和软件级部署。

计算子系统：总体而言，EVOLVE 中使用的 HPC 集群

会议日期：2022 年 3 月 14-23 日
会议地点：比利时安特卫普
添加到 IEEE 的日期 *Xplore*：2022 年 5 月 19 日
发布者：IEEE
电子 ISBN：978-3-9819263-6-1
按需打印 (PoD) ISBN：978-1-6654-9637-7
电子 ISSN： 1558-1101
按需打印 (PoD) ISSN： 1530-1591

DOI：10.23919/DATE54114.2022.9774698
第四卷(期):975-980,2022 年

结合了 16 个异构 Intel x86 计算节点（Broadwell、Haswell 和 Skylake 系列），并安装了各种加速器和存储技术，如表 1 所述。这里所说的加速器，即把 GPU 和 FPGA 安装在其中六个节点上，所有节点都通过 NVIDIA Mellanox InfiniBand FDR 链路（56 Gb/s）互连并在 Linux 运行。

存储子系统：该平台使用不同类型的存储：网络文件系统为方便起见提供主目录，而 Lustre 和 IME [2]则集成了高性能 I/O 操作。IME 提供高性能，允许 I/O 密集型操作，重点关注带宽和大量 I/O 数据请求。高度并行性与快速闪存器件相结合，可提供最高级别的性能。IME 充当 Lustre 存储池的突发缓冲区，它提供实际容量。

计算能力：为了评估 EVOLVE 的硬件基础设施的性能，我们利用不同的开源和内部开发的基准。具体来说，对于大数据工作负载，我们使用 HiBench 基准测试套件[3]，它提供了一组 Spark 内存应用程序。关于 HPC 工作负载，我们使用了高性能共轭因子（HPCG）[4]基准，它支持 MPI，OpenMP 和 CUDA，旨在练习在现代机器学习和深度学习工作负载中广泛使用的计算和数据访问模式。最后，为了测试平台的加速能力，我们为目标设备开发了两个内部版本的 VGG16 和 ResNet50 推理模型[5]。

TABLE I: Advanced Computing Platform

表 1：高级计算平台

CPUs	Intel Xeon {Platinum 8153, E5-2690/2670/2470}
GPUs	Nvidia {Tesla V100, P40, K20Xm}
FPGAs	Altera Arria 10, Stratix 10
Storage	IME, Lustre, NFS

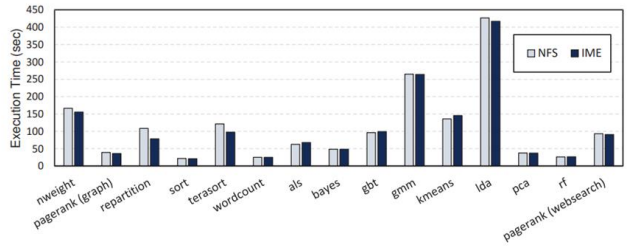


Fig. 2: Storage Capabilities of EVOLVE's HW infrastructure over HiBench benchmark suite

图 2：EVOLVE 的硬件基础设施在 HiBench 基准套件上的存储能力

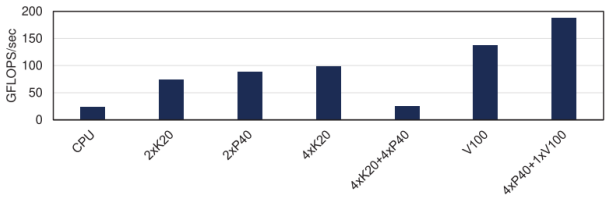


Fig. 3: Computing Capacity of EVOLVE's HW infrastructure over HPCG benchmarks (CPU+GPU)

图 3：EVOLVE 的硬件基础设施在 HPCG 基准测试中的计算能力（CPU+GPU）

图 2 和图 3 以及表 2 展示了各自的性能结果。在图 2 中，我们展示了不同的数据密集型基准测试的执行时间，当它们使用 IME 和 NFS 存储子系统时。如图所示，对于其中的大多数，IME 存储提供了平均 5%的性能改进，预计将进一步提高并发和协同部署场景。关于 EVOLVE 堆栈的计算能力，图 3 评估了不同基础设施组成下每秒实现的 GFLOPS。我们可以清楚地看到，GPU 加速器大大提高了集群的计算能力，与仅使用 CPU 的版本相比，吞吐量（GFLOPS/秒）提高了 4 倍。最后，表 2 展示了，在我们的硬件基础设施中，两个 CNN 对于不同实现和安装条件下的目标设备的执行时间。如图所示，GPU 和 FPGA 的使用产生了 2 倍到 11 倍的加速比。

TABLE II: Evaluation of HW accelerators on CNN models.

表 2：CNN 模型上 HW 加速器的评估。

Platform	Execution time (ms)	
	VGG16	ResNet50
TF Intel Xeon Platinum 8153	1067.26	1162.79
TF Nvidia K20Xm	1095.19	410.25
TF Nvidia P40	608.35	187.71
TF Nvidia V100	318.24	143.83
OpenCL Stratix10	103.1	101.21

3 EVOLVE 软件栈

EVOLVE 提供两个层次的软件栈部署，即系统级部署，旨在跨平台和软件级部署提供透明的优化执行和存储功能，向最终用户提供用户友好的端点。

系统级部署：我们的平台使用 Kubernetes 编排器，用于无缝的容器化应用程序部署。我们的软件栈通过集成火山[6]进行批量调度，并开发两个自定义 Kubernetes 调度器，扩展了 Kubernetes 的资源管理方案。首先，我们的干扰感知调度器[7]将传入的应用程序放在可用资源池中。要做到这一点，它根据节点所经历的资源争用来优先考虑节点[8]，[9]。其次，Skynet [10]提供实时资源调优，基于执行期间收集的指标和用户指定的目标性能指标动态调整应用程序运行时配置文件和资源分配。此外，为了科普可用硬件提供的冲突存储抽象，并

DOI：10.23919/DATE54114.2022.9774698
第四卷(期):975-980,2022 年

解决与数据交互的 API 和编程库的异构性，我们设计并实现了统一存储层（USL）[11]。Datashim [12]是 USL 的核心，它将实际的数据集装载到容器中，从而统一了对各种实际存储协议和技术的访问。Finalt，H3 [13]是一个高性能的对象存储，可以被应用程序直接使用（作为库嵌入），或者通过基于 FUSE 的兼容层挂载为 USL 数据集。

软件级部署：软件框架作为工作流执行的运行时组件，被打包在容器中作为微服务，然后用作工作流步骤的构建块。大多数微服务处理计算密集型任务，而一些微服务实现支持服务。微服务包括：Kafka，具有自动调优功能的 Spark [14]，用于监控的 TensorFlow，MPI，Dask，Promethes 和 Grafana，以及 CI/CD 服务，如 Jenkins 和 Git 存储库。用户通过 Karvdash [15]（EVOLVE 仪表板）与平台交互，并使用与平台微服务和软件堆栈的可视化组件接口的工作流在笔记本电脑中实现他们的应用程序。Karvdash 允许从预定义的模板在容器中编排服务执行，通过使用 USL 与数据集合进行交互，并在一个外部可访问的端点下安全地提供多个服务。

特别是对于 HPC，EVOLVE 使工作流程能够无缝地将基于 MPI 的可执行文件作为处理阶段，从而朝着 HPC 和 BigData 处理的集成迈出了具体的一步。我们部署按需虚拟集群-基于容器的 HPC 环境，允许 MPI 代码在 Kubernetes 中运行，使用 Slurm 作业管理器和各种库和编译器，同时保持对物理节点上存在的 InfiniBand 网络以及可用 GPU 和 FPGA 的直接访问。每个虚拟集群都有一个自定义的 Slurm 控制器，该控制器与 Skynet 通信，以协调 HPC 作业的实际放置。

TABLE III: Technologies used per use-case

表 III：每个用例使用的技术

Technologies/Pilots	❶	❷	❸	❹	❺
IME Fast Storage	✓	✓	✓	✓	✓
GPU/FPGA Acceleration	✓	✓	✓	✓	✓
Resource- & Interference-aware Orchestration		✓		✓	
Big Data Processing		✓	✓		
Scale-out parallel frameworks	✓	✓	✓	✓	✓
One-Pass stream engines			✓	✓	
Prometheus/Grafana Monitoring	✓			✓	

4 试点工作流的部署和评估

EVOLVE 试验台通过一组不同的用例进行评估。在本文中，我们重点讨论了五个应用领域，在第 4.1 节中的①-⑤中进行了详细说明。表 3 显示了所检查的用例与所使用的相应 EVOLVE 技术之间的映射，如第 3 节所述。

4.1 试点说明和评估

①使用基于知识图的自动驾驶数据集成进行目标检测：该用例实现了计算机视觉方法，使用车辆仪表盘视频检测两个驾驶员辅助系统（车道辅助和自适应巡航控制）的状态，以评估自动驾驶的接受程度。使用基于知识图的数据集成框架[16]，我们部署了对象检测概念验证。知识图用于将存储在单独数据源中的联合收割机数据组合成单个统一视图[17]。我们将该框架应用于广为人知的 Motional 和 Lyft 数据集，以评估使用 EVOLVE 提供的高性能集成计算平台的好处。这些服务是通过利用 Karvdash 和 Datashim 支持的集成 USL 开发和部署的。还使用了 Argo 工作流，以便在 Kubernetes 的上下文中实现无缝并行执行。

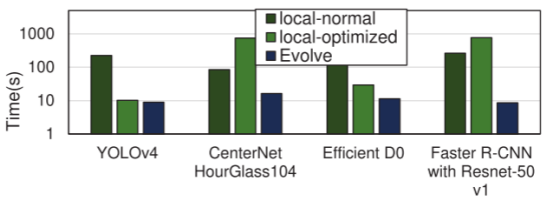


Fig. 4: Object detection models execution time comparison

图 4：对象检测模型执行时间比较

评估：通过 EVOLVE 平台，我们能够在 2101.52 秒内为 Motional 创建 6,538,057 个节点，在 445.6 秒内为 Lyft 创建 1,047,118 个节点。与 EVOLVE 之前的基础设施在 22.7 秒内交付 82,454 个节点和在 4.14 秒内交付 9,442 个节点相比，EVOLVE 允许利用更大的知识图，从而产生更有效的对象检测模型。此外，图 4 比较了不同对象检测模型的顺序执行与在 EVOLVE 平台上针对 1000 个图像的并行执行。更具体地说，我们绘制正常和优化的本地，顺序版本，旁边的 EVOLVE 的并行。EVOLVE 的基础设施为更快的 R-CNN 提供了高达 30 倍的加速，Resnet-50 v1 和 CenterNet HourGlass 104 分别为 5 倍。

②使用观察和历史运营数据改善公交运输服务：该用例旨在基于后分析活动（非真实的时间上下文）和公共交通网络上的当前交通拥堵（真实的时间上下文）连续评估公交服务质量。为此，计划数据与实际数据相关联，包括与框架条件（交通，天气，需求）相对应的数据。关于非实时（NRT）环境，分析需要应用于更长的时间段，在可接受的响应时间内以许多不同的方式可视化。另一方面，实时（RT）上下文的情况下，涉及到延迟的情况下，注意到和相应的可视化的识别。在这两种情况下，需要将大型数据集摄取到系统中，有效地进行转换和可视化。

从数据摄取到 Spark 处理和可视化的端到端管道已集成到 EVOLVE 平台。需要扩大当局和运营商在其决策和规划程序中使用的批量数据分析的范围，同时能够对网络中的总线事

DOI：10.23919/DATE54114.2022.9774698
第四卷(期):975-980,2022 年

件数据进行实时分析。因此，在不同的设置下考虑两种不同的工作流：

- 离线活动工作流程，旨在从运营商和管理局的角度，根据从由 PT 公司和/或当局直接管理的 ITS 系统收集的认证和验证数据，改善计划的公共交通（PT）服务。
- 真实的时间活动（RT）旨在给予交通网络上的交通拥塞发生时的信息。这些数据对于管理机构和操作员都是有用的，以便执行真实的实时监控活动并向最终用户提供关于最佳旅行计划的信息。

这两个数据工作流分别利用 IME、Spark 与 Spark 流引擎的结合，以及 Apache Zeppelin over EVOLVE 来改善公共交通运营中的决策支持和主动监控。

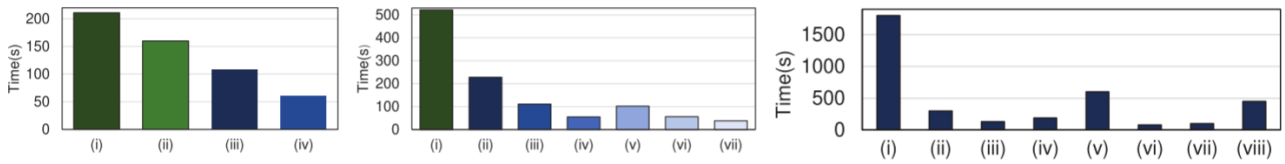
评估：通过 EVOLVE，我们在 20 秒内同时处理和可视化了 21 k 个弧线。这使得 i）增加了分析所涵盖的时间段，从一个月的数据（220 个巴士班次）到一年以上的时间段。ii）加快两天服务的行程列表，iii）减少汇总过境时间的日常查询的延迟，iv）在不到 6 秒的时间内完成两到三条选定线路/路线的实时状态的可视化过程。最后，现在可以在 PT 道路网络的地图上可视化交通拥堵，并可以突出显示拥堵路段。

3.利用观测数据、历史元数据和分类模型进行海事监测：海事监测已被确定为欧盟一级的首要任务。通常通过将来自 SAR（合成孔径雷达）图像的船舶信息与 AIS/VHF 数据相关联来解决这一挑战。所需的计算密集型处理，今天强加了很长的响应时间，这种方法，这降低了其操作价值。

简单的并行化和分布式计算只能部分解决这个问题，这是由于在多个计算节点上划分卫星图像的固有限制; HPC 和硬件加速需要发挥作用，以产生更好的结果。海上监视工作流程的目的是简化和自动化整合和增强海上情况图片所需的流程。工作流程包括以下阶段：AIS（自动识别系统）数据采集，AIS 异常检测，SAR 图像采集，SAR 滤波/预处理和血管检测，AIS/SAR 融合相关，最后，可视化。工作流程的所有阶段都部署在 EVOLVE 平台中，并通过 Zeppelin 前端进行管理。为了自动检测可能存在的血管和任何异常，使用 Argo 工作流。创建了两个 SQL 查询，用于根据位置和时间过滤卫星图像和 AIS 数据，以便两个并行工作流运行。

选择时间窗口并执行血管检测算法：第一个查询返回符合用户定义标准的卫星图像（SAR）。请注意，有两个 cron 作业定期运行并下载 SAR 图像（到 EVOLVE Kubernetes 集群上的持久卷），同时还存储每个图像的元数据，例如，时间，地理位置，到关系数据库成功执行查询后，对于找到的每个图像，在 EVOLVE 平台上触发一个新的 pod，每个 pod 并行搜索图像中的血管，保存结果并将其转换为 GeoJson 格式进行可视化。

评估：首先，我们评估了处理所需的时间，即，船舶检测和分类，在一个 SAR 场景。这被计算为连续处理的 10 个 SAR 场景的平均处理时间。我们评估了四种替代的增量部署方案：i）具有串行处理的遗留系统; ii）具有串行处理的 Argo 工作流，本地; iii）具有串行处理的 Argo 工作流，iv）EVOLVE 集群; v）具有并行处理的 Argo 工作流，EVOLVE 集群。图 5a 描绘了跨四种场景的处理时间的改进。总的来说，与传统配置相比，由于粗粒度并行化，EVOLVE 技术和基础设施迄今为止已经大大减少了处理时间（71%）。



(a) SAR scene processing (a)SAR 场景处理

(b) Anomaly detection based on AIS data (b)基于 AIS 数据的异常检测

(c) Change detection (c)变化检测

Fig. 5: Maritime surveillance (a), (b) and Change detection on satelite images (c). Indexes (i)-(viii) refer to the mentioned differing configuration scenarios.

图 5：海上监视(a)、(b)和卫星图像变化检测（c）。索引(i)-(viii)是指所提及的不同配置场景。

我们还测量了 AIS 异常检测算法的训练时间，该算法用于检测船舶轨迹中的可疑行为。使用的异常检测算法具有自动编码器的结构，由 6 个卷积层组成（编码器的 3 个 CNN 和解码器的 3 个 CNN）。该算法已经训练了 500 个 epoch，批量大小等于 4096。训练集大约有 370 万条记录，其中 10%用作验证集。经过一些预处理后，22 列已用于训练。下面的图 5 b 显示了所有不同部署的培训时间的改进。评估的场景是 i）遗留系统，ii）EVOLVE CPU，iii）EVOLVE Spark，iv）EVOLVE Spark-GPU，v）GPU-K20 Xm，vi）GPU P40 和 vii）GPU Tesla V100。总的来说，与第一种情况（单台机器中的单 CPU）相

DOI：10.23919/DATE54114.2022.9774698
第四卷(期):975-980,2022 年

比，EVOLVE 集群基础设施迄今为止已经在最坏情况下（当 EVOLVE 平台中仅使用 CPU 时）将训练时间大幅减少了 66.3%，在使用非常强大的 GPU（Tesla V100）的最佳情况下达到了 92.7%。

最后，我们评估的 AIS 异常检测的准确性。对于此评估，自动编码器的架构已更改。利用 EVOLVE 集成集群的巨大内存容量，我们将 AIS 记录的批次从 4，096 增加到 204，800，五层 Conv2D 架构从 32x16x8x16x32 增加到 512x256x128x256x512。前一个模型的准确率为 84%，损失为 0.1，而在 EVOLVE 平台中部署更新的配置后，我们将准确率提高到 89%，损失相同。

④卫星图像的辐射校正和变化检测：随着哥白尼计划及其丰富的开放数据的出现，地球观测领域越来越多地采用大数据技术。高效数据存储和处理基础设施的出现，使得能够开发允许自动或半自动分析大量地球观测数据的应用程序，如从高地理尺度识别模式或从多时相数据集提取长期趋势。在多时态变化检测中的一个重要挑战是快速访问和存储大量的数据和计算密集型的处理，这是必要的。在这种情况下，使用 EVOLVE 平台，其中包含 HPC 功能，是非常有用的。该试点项目的目标是检测整个欧洲在一年内的变化，重访期为 10 天。因此，此更改检测应用程序需要访问和处理大约 35，000 个 Sentinel-2 数据的磁贴（20 TB 存储）。

评估：我们评估了这个应用程序的 8 个不同版本：i）仅 CPU，ii）仅 CPU 使用 TensorFlow（TF），iii）GPU 和 TF，iv）DASK，v）DASK 和 GPU，vi）小数据集中的 Kafka 和 DASK，vii）小数据集中的 Kafka 和 GPU，viii）大数据集中的 Kafka 和 DASK。DASK 和 Kafka 技术允许自动化流水线，并在不同模块之间进行并行计算。Argo 为 Kubernetes 集群的编排做出了贡献，而分别由 Prometheus 和 Grafana 启用的监控服务和资源可视化提供了有关资源使用的有用见解。图 5c 显示了 1）DASK 调度器将计算时间减少了十倍，2）GPU 加速器也显著减少了计算时间，3）DASK 与 GPU 的组合在延迟中引起了不可忽略的开销，以及 4）Kafka 流引擎与 DASK 的组合也减少了每个变化检测图的平均计算时间。

⑤数据辅助汽车服务开发：使用大型发动机状态监测数据说明时间序列模式检测。在这种情况下，气缸压力曲线是提供时间序列数据的重要来源。它包含有关所有组件及其状况的信息。结合其他发动机运行数据并通过使用对时间序列数据进行操作的专用算法，可以检测许多故障症状和相应的根本原因信息。模式检测和匹配是时间序列测量的必要工作流程中最关键的功能之一。模式与表征正常行为的许多模式相关，然后才能将特定模式归类为正常或有问题。就计算资源而言，这是一个非常消耗的过程，同时要求在一秒钟内提供响应。

在这个试点中有两个用例。第一个是计算密集型应用程序，执行时间序列分析以进行异常检测。来自许多不同通道的时间序列被累积、预处理、转换，最后模式检测算法将每个时间序列与定义每个通道的正常行为的参考曲线进行比较。另一个是数据量密集型应用程序，运行大量汽车测量数据，主要执行描述性统计。

评估：通过利用 Argo 工作流在 Kubernetes 上无缝并行化和部署应用程序，我们成功地在第一个用例中显著提高了性能，如图 6 所示，在第二个用例中提高了 40-50%。然而，GPU 的利用率并没有提供任何改善的执行时间。GPU 实现是在执行矩阵乘法的小部分代码上实现的，这部分代码对整体执行时间的贡献微不足道。事实上，利用 Prometheus 分析服务，我们表明，特定的功能已经贡献了不到 1%的总执行时间。

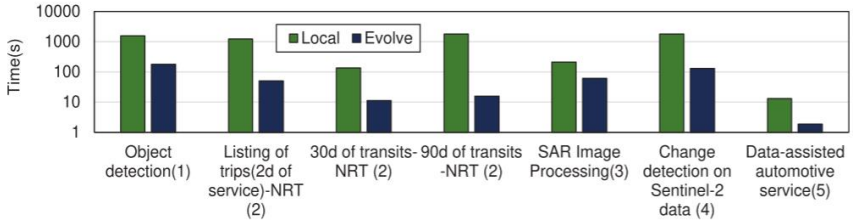


Fig. 6: Legacy and EVOLVE platform comparison on various use-cases

图 6：Legacy 和 EVOLVE 平台在各种用例上的比较

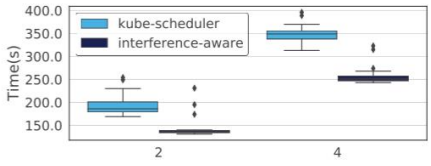


Fig. 7: Interference-aware orchestrator's impact on latency distribution

图 7：干扰感知协调器对延迟分布的影响

4. 2 综合试验台的总体影响

EVOLVE 平台在多个用例的开发中实现了优化或显着提高了性能。与传统部署相比，EVOLVE 技术实现了平均延迟 26.3 倍的改进。

更具体地说，对于目标检测试验中 100 个图像的延迟，如图 6 所示，我们实现了传统系统部署的 8.73 倍加速。然后，在图 6 中呈现了公共交通工作流 2 的三个不同阶段的执行时间，即，两天服务的清单（24.7 倍加速），30 天和 90 天的过境（分别为 12 倍和 114 倍加速）。SAR 图像处理的空中监视飞行员提高了 3.45 倍，而变化检测的哨兵-2 数据呈现减少延迟 13.84 倍。最后，对于汽车服务飞行员，虽然每缸延迟较高，由于集装箱化引起的延迟，我们总共实现了 7 倍的加速。

最后，EVOLVE 在现实的重负载 I 场景下保持了其效率。图 7 显示了当 EVOLVE 节点上应用各种级别的资源干扰时，干扰感知调度器对导频②和导频②的影响。如图 7 所示，EVOLVE 的协调器识别并防止在竞争节点上分配作业;因此与

DOI：10.23919/DATE54114.2022.9774698
第四卷(期):975-980,2022 年

Kubernetes 的默认调度器相比，实现了更高的性能（平均 26%），这也导致按比例降低了能耗。

5 结论

EVOLVE 是一个多合作伙伴协作计划，使我们更接近 HPC，云计算和大数据世界的融合。先进的计算平台与项目中实施的系统和软件级技术相结合，为紧密集成的 EVOLVE 测试平台做出了贡献。因此，HPC 功能、透明度、可移植性和大规模数据分析使来自不同领域的试点能够在融合平台上构建、移植、部署和优化其用例。通过采用 EVOLVE 技术，我们实现了平均 26.3 倍的性能提升，在整个测试用例集上达到了 114 倍。

参 考 文 献

[1] Chazapis, Antony , et al. "EVOLVE: HPC and cloud enhanced testbed for extracting value from large-scale diverse data." CF '21: Computing Frontiers Conference 2021.

[2] “Ddn. infinite memory engine,” https://www.ddn.com/products/ime-flash-native-data-cache/

[3] S. Huang, J. Huang, J. Dai, T. Xie, and B. Huang, “The hibench benchmark suite: Characterization of the mapreduce-based data analysis,” in 2010 IEEE 26th International Conference on Data Engineering Workshops (ICDEW 2010). IEEE, 2010, pp. 41–51.

[4] J. Dongarra, M. A. Heroux, and P. Luszczek, “Hpcg benchmark: a new metric for ranking high performance computing systems,” Knoxville, Tennessee, p. 42, 2015.

[5] K. Koliogeorgi, F. E. Keddous, D. Masouros, A. Chazapis, M. Aubrun, S. Xydis, A. Bilas, R. Hugues, J.-T. Acquaviva, H. N. Nguyen et al., “Fpga acceleration in evolve’s converged cloud-hpc infrastructure,” in 2021 31st International Conference on Field-Programmable Logic and Applications (FPL). IEEE, 2021, pp. 376–377.

[6] “A kubernetes native batch system,” https://github.com/volcano-sh/ volcano.

[7] A. Tzenetopoulos, D. Masouros, S. Xydis, and D. Soudris, “Interferenceaware orchestration in kubernetes,” in International Conference on High Performance Computing. Springer, 2020, pp. 321–330.

[8] ——, “Interference-aware workload placement for improving latency distribution of converged hpc/big data cloud infrastructures.”

[9] D. Masouros, S. Xydis, and D. Soudris, “Rusty: Runtime interferenceaware predictive monitoring for modern multi-tenant systems,” IEEE Transactions on Parallel and Distributed Systems, vol. 32, no. 1, pp. 184–198, 2020.

[10] Y. Sfakianakis, C. Kozanitis, C. Kozyrakis, and A. Bilas, “Quman: Profile-based improvement of cluster utilization,” ACM Transactions on Architecture and Code Optimization (TACO), vol. 15, no. 3, pp. 1–25, 2018.

[11] A. Chazapis, C. Pinto, Y. Gkoufas, C. Kozanitis, and A. Bilas, “A unified storage layer for supporting distributed workflows in kubernetes,” in Proceedings ofthe Workshop on Challenges and Opportunities ofEfficient and Performant Storage Systems, 2021, pp. 1–9.

[12] https://github.com/datashim-io/datashim“Datashim: A kubernetes based framework for hassle free handling of datasets,”

[13] A. Chazapis, E. Politis, G. Kalaentzis, C. Kozanitis, and A. Bilas, “H3: An application-level, low-overhead object store,” in High Performance Computing, H. Jagode, H. Anzt, H. Ltaief, and P. Luszczek, Eds. Cham: Springer International Publishing, 2021, pp. 174–188.

[14] D. Nikitopoulou, D. Masouros, S. Xydis, and D. Soudris, “Performance analysis and auto-tuning for spark in-memory analytics,” in 2021 Design, Automation & Test in Europe Conference & Exhibition (DATE). IEEE, 2021, pp. 76–81.

[15] “Karvdash: A dashboard service for facilitating data science on kubernetes,” https://github.com/CARV-ICS-FORTH/karvdash.

[16] T. E. Kalaycı, B. Bricelj, M. Lah, F. Pichler, M. K. Scharrer, and J. Rubeˇsa-Zrim, “A knowledge graph-based data integration framework applied to battery data management,” Sustainability, vol. 13, no. 3, p. 1583, 2021.

[17] T. E. Kalayci, E. G. Kalayci, G. Lechner, N. Neuhuber, M. Spitzer, E. Westermeier, and A. Stocker, “Triangulated investigation of trust in automated driving: Challenges and solution approaches for data integration,” Journal of Industrial Information Integration, vol. 21, p. 100186, 2021